

# Modeling and Simulation for Physical Vapor Deposition: Multiscale Model

Jürgen Geiser and Robert Röhle

Humboldt-Universität zu Berlin,  
Department of Mathematics,  
Unter den Linden 6, D-10099 Berlin, Germany  
`geiser@mathematik.hu-berlin.de`

**Abstract.** In this paper, we present modeling and simulation for physical vapor deposition for metallic bipolar plates.

In the models, we discuss the application of different models to simulate the transport of chemical reactions of the gas species in the gas-chamber. The so called sputter process is an extremely sensitive process to deposit thin layers to metallic plates.

We taken into account lower order models to obtain first results with respect to the gas fluxes and the kinetics in the chamber.

The model equations can be treated analytically in some circumstances and complicate multi-dimensional models are solved numerically with a software-packages (UG unstructured grids, see [2]).

Because of multi-scaling and multi-physical behavior of the models, we discuss adapted schemes to solve more accurate in the different domains and scales.

The results are discussed with physical experiments to give a valid model for the assumed growth of thin layers.

**Keywords:** Physical vapor deposition, multi-scale problem, convection-diffusion equations, reaction equations, splitting methods.

**AMS subject classifications.** 35K25, 35K20, 74S10, 70G65.

## 1 Introduction

We motivate our studying on simulating a thin film deposition process that can be done with PVD (physical vapor deposition) processes. In the last years, the research of producing thin films to metallic plates have been increased. Novel deposition methods are low temperature and low pressure processes, that can be controlled by an underlying plasma, see [3], [1]. One of the interests on standard applications to TiN and TiC are immense but recently also deposition processes with new material classes known as MAX-phases are important. The MAX-phase are nanolayered ternary metal-carbides or -nitrides, where  $M$  is a transition metal,  $A$  is an A-group element (e.g. Al, Ga, In, Si, etc.) and  $X$  is  $C$  (carbon) or  $N$  (nitride). Such materials with nanolayered MAX-phase films can be used in the

production of metallic bipolar plates, where the new thin film is at least non-corrosive and a metallic conductor.

We present models, that can be used to control flow and transport of gaseous species to the deposition layer, see [1] and [21].

We deal with a continuous flow model, while we assume a vacuum and diffusion dominated process. The models can be solved with convection-diffusion equations. Further, we deal with kinetic models to understand the material fluxes in the PVD processes, see [3].

To solve the model equations, we use analytical as also numerical methods to be efficient as possible in the solver process, see [13].

To be precise, for numerical methods, we apply finite volume discretizations for the spatial terms and the backward Euler method or Crank-Nicolson method for the time discretization.

To couple analytical and numerical solvers together, we apply operator splitting methods, as effective coupling schemes. Such splitting methods can be seen as microscopic decoupling schemes to understand complicate mixed physical effects, e.g. flux streams of the species, reactions between the species and retardation processes. This can be helped to discuss each dominant physical effect in a separate decoupled model, see [14].

The paper is outlined as follows.

In section 2 we present our mathematical model and a possible reduced model for the further approximations. To solve our model equations, we apply various analytical and numerical methods, which are presented in Section 3. The decomposition methods to separate the singular and non-singular reaction systems are explained in Section 4. The numerical experiments are given in Section 5.

In the contents, that are given in Section 6, we summarize our results.

## 2 Mathematical Model

In the following, the models are discussed in two directions of far-field and near-field problems:

1. Reaction-diffusion equations, see [15] (far-field problem);
2. Boltzmann-Lattice equations, see [21] (near-field problem).

The modeling is considered by the Knudsen Number ( $Kn$ ), which is the ratio of the mean free path  $\lambda$  over the typical domain size  $L$ . For small Knudsen Numbers  $Kn \approx 0.01 - 1.0$ , we deal with a Navier-Stokes equation or with the Convection-Diffusion equation, see [16] and [18], whereas for large Knudsen Numbers  $Kn \geq 1.0$ , we deal with a Boltzmann equation, see [19].

### 2.1 Model for Small Knudsen Numbers (Far Field Model)

When gas transport is physically more complex because of combined flows in three dimensions, the fundamental equations of fluid dynamics become the starting point of the analysis. For our models with small Knudsen numbers, we can

assume a continuum flow, and the fluid equations can be treated with a Navier-Stokes or especially with a convection-diffusion equation.

Three basic equations describe the conservation of mass, momentum and energy, that are sufficient to describe the gas transport in the reactors, see [19].

1. Continuity : The conservation of mass require that the net rate of the mass accumulation in a region be equal to the difference between the inflow and outflow rate.
2. Navier-Stokes : Momentum conservation requires that the net rate of momentum accumulation in a region be equal to the difference between the in and out rate of the momentum, plus the sum of the forces acting on the system.
3. Energy : The rate of accumulation of internal and kinetic energy in a region is equal to the net rate of internal and kinetic energy in by convection, plus the net rate of heat flow by conduction, minus the rate of work done by the fluid.

We will concentrate us to the conservation of mass and assume that the energy and momentum is conserved, see [15]. Therefore the continuum flow can be described as convection-diffusion equation given as:

$$\frac{\partial}{\partial t}c + \nabla F - R_g = 0, \text{ in } \Omega \times [0, T] \quad (1)$$

$$F = -D\nabla c,$$

$$c(x, t) = c_0(x), \text{ on } \Omega, \quad (2)$$

$$c(x, t) = c_1(x, t), \text{ on } \partial\Omega \times [0, T], \quad (3)$$

where  $c$  is the molar concentration and  $F$  the flux of the species.  $D$  is the diffusivity matrix and  $R_g$  is the reaction term. The initial value is given as  $c_0$  and we assume a Dirichlet Boundary with the function  $c_1(x, t)$  sufficient smooth.

## 2.2 Model for Large Knudsen Numbers (Near Field Model)

The model assumes that the heavy particles can be described with a dynamical fluid model, where the elastic collisions define the dynamics and few inelastic collisions are, among other reasons, responsible for the chemical reactions.

To describe the individual mass densities as well as the global momentum and the global energy as dynamic conservation quantities of the system, corresponding conservation equations are derived from Boltzmann equations.

The individual character of each species is considered by mass-conservation equations and the so-called difference equations.

The Boltzmann equation for heavy particles (ions and neutral elements) is now given as

$$\frac{\partial}{\partial t}n_s + \frac{\partial}{\partial \mathbf{r}} \cdot (n_s \mathbf{u} + n_s c_s) = Q_n^{(s)}, \quad (4)$$

$$\begin{aligned} \frac{\partial}{\partial t} \rho \mathbf{u} + \frac{\partial}{\partial \mathbf{r}} \cdot (\rho \mathbf{u} \mathbf{u} + n T \underline{\underline{I}} - \underline{\underline{\tau}}^*) &= \sum_{s=1}^N q_s n_s \langle \mathbf{E} \rangle, \\ \frac{\partial}{\partial t} \mathcal{E}_{\text{tot}}^* + \frac{\partial}{\partial \mathbf{r}} \cdot (\mathcal{E}_{\text{tot}}^* \mathbf{u} + \mathbf{q}^* + n T \mathbf{u} - \underline{\underline{\tau}}^* \cdot \mathbf{u}) &= \sum_{s=1}^N q_s n_s (\mathbf{u} + \mathbf{c}_s) \cdot \langle \mathbf{E} \rangle - Q_{\mathcal{E}, \text{inel}}^{(e)}, \end{aligned}$$

where  $\rho$  denotes the mass density,  $\mathbf{u}$  is the velocity, and  $T$  the temperature of the ions.  $\mathcal{E}_{\text{tot}}^*$  is the total energy of the heavy particles.  $n_s$  is the particle density of heavy particle species  $s$ .  $\mathbf{q}^*$  is the heat flux of the heavy particle system.  $\underline{\underline{\tau}}^*$  is the viscous stress of the heavy particle system.  $\mathbf{E}$  is the electric field and  $\underline{\underline{Q}}_{\mathcal{E}}$  is the energy conservation.

Further the production terms are  $Q_n^{(s)} = \sum_r a_{\text{sign}} k_{\alpha, r} n_{\alpha} n_r$  with the rate coefficients  $k_{\alpha, r}$ .

We have drift diffusion for heavy particles in the following fluxes. The dissipative fluxes of the impulse and energy balance are linear combinations of gene-realized forces,

$$\begin{aligned} \mathbf{q}^* &= \lambda_E \langle \mathbf{E} \rangle - \lambda \frac{\partial}{\partial \mathbf{r}} T - \sum_{s=1}^N \sum_{\alpha=1}^N \lambda_n^{(\alpha, s)} \frac{1}{n_s} \frac{\partial}{\partial \mathbf{r}} n_{\alpha}, \\ \underline{\underline{\tau}}^* &= -\eta \left( \frac{\partial}{\partial \mathbf{r}} \mathbf{u} + \left( \frac{\partial}{\partial \mathbf{r}} \mathbf{u} \right)^{\top} - \frac{2}{3} \left( \frac{\partial}{\partial \mathbf{r}} \cdot \mathbf{u} \right) \underline{\underline{I}} \right), \\ \mathcal{E}_{\text{tot}}^* &= \sum_{s=1}^N 1/2 \rho_s c_s^2 + 1/2 \rho u^2 + 3/2 n T. \end{aligned}$$

where  $\lambda$  is the thermal diffusion transport coefficient.  $T$  is the temperature,  $n$  is the particle density.

Diffusion of the species are underlying to the given plasma and described by the following equations

$$\begin{aligned} \frac{\partial}{\partial t} n_s + \frac{\partial}{\partial \mathbf{r}} \cdot (n_s \mathbf{u} + n_s \mathbf{c}_s) &= Q_n^{(s)}, \\ \mathbf{c}_s &= \mu_s \langle \mathbf{E} \rangle - d_T^{(s)} \frac{\partial}{\partial \mathbf{r}} T - \sum_{\alpha=1}^N D_n^{(\alpha, s)} \frac{1}{n_s} \frac{\partial}{\partial \mathbf{r}} n_{\alpha}. \end{aligned}$$

The density of the species are dynamical values and the species transport and mass transport are underlying to the following constraint conditions:

$$\begin{aligned} \sum_s m_s n_s &= \rho, \\ \sum_s n_s m_s \mathbf{c}_s &= 0. \end{aligned}$$

where  $m_s$  is the mass of the heavy particle,  $n_s$  is the density of the heavy particle,  $\mathbf{c}_s$  is the difference velocity of the heavy particle.

### Field Model

The plasma transport equations are maxwell equations and are coupled with a field. They are given as

$$\frac{1}{\mu_0} \nabla \times \mathbf{B}_{\text{dyn}} = -e n_e \mathbf{u}_e + \tilde{\mathbf{j}}_{\text{ext}}, \quad (5)$$

$$\nabla \cdot \mathbf{B}_{\text{dyn}} = 0, \quad (6)$$

$$\nabla \times \mathbf{E} = -\frac{\partial}{\partial t} \mathbf{B}_{\text{dyn}}, \quad (7)$$

where  $\mathbf{B}$  is the magnetic field and  $\mathbf{E}$  is the electric field.

### 2.3 Simplified Model for Large Knudsen Numbers (Near Field Model)

For the numerical analysis and for the computational results, we reduce the complex model and derive a system of coupled Boltzmann and diffusion equations.

We need the following assumptions:

$$\begin{aligned} \mathbf{q}^* &= -\lambda \frac{\partial}{\partial \mathbf{r}} T, \\ \underline{\underline{T}}^* &= 0, \\ \mathcal{E}_{\text{tot}}^* &= 3/2nT, \\ Q_{\mathcal{E},\text{inel}}^{(e)} &= \text{const}, \end{aligned}$$

and obtain a system of equations:

$$\begin{aligned} \frac{\partial}{\partial t} \rho + \frac{\partial}{\partial \mathbf{r}} \cdot (\rho \mathbf{u}) &= 0, \\ \frac{\partial}{\partial t} \rho \mathbf{u} + \frac{\partial}{\partial \mathbf{r}} \cdot (\rho \mathbf{u} \mathbf{u} + nT \underline{\underline{I}}) &= \sum_{s=1}^N q_s n_s \langle \mathbf{E} \rangle, \\ \frac{\partial}{\partial t} 3/2nT + \frac{\partial}{\partial \mathbf{r}} \cdot \left( 3/2nT \mathbf{u} + \lambda \frac{\partial}{\partial \mathbf{r}} T + nT \mathbf{u} \right) &= \sum_{s=1}^N q_s n_s (\mathbf{u} + \mathbf{c}_s) \cdot \langle \mathbf{E} \rangle - Q_{\mathcal{E},\text{inel}}^{(e)}. \end{aligned}$$

*Remark 1.* We obtain three coupled equations for the density, velocity and the temperature of the plasma. The equations are strong coupled and a decomposition can be done in the discretized form.

## 3 Discretization methods

In this section, we deal with the discretization methods that we use to discretize ODE and PDE systems.

### 3.1 Analytical Solution to the systems of ordinary differential equations of first order

The following ODE system is given :

$$\frac{dc}{dt} = A c(t) + b(t), \quad (8)$$

where  $c(t = t_0) = c_0$ .

We assume the matrix  $A$  is constant and non-singular. Further we assume that  $b(t)$  is a smooth function of  $t$ .

For the solution, we obtain the method of integrating factor and to a transformation to eigenvalues.

We have

$$c(t) = c_{\text{homo}}(t) + c_p(t) , \quad (9)$$

where  $c(t = t_0) = c_0$ .

For the homogeneous part we have:

$$c_{\text{homo}}(t) = W_c \exp(\Lambda(t - t_0)) \tilde{c} , \quad (10)$$

where  $\tilde{c} = W_c^{-1} c_0$  and

$$\exp(\Lambda(t - t_0)) = \begin{pmatrix} \exp(\lambda_1(t - t_0)) & 0 & \dots & 0 \\ 0 & \exp(\lambda_2(t - t_0)) & \dots & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & \dots & \exp(\lambda_n(t - t_0)) \end{pmatrix} , \quad (11)$$

For the inhomogeneous part we have:

$$c_p(t) = W_c \exp(\Lambda(t)) u(t) , \quad (12)$$

where  $u(t) = \int_{t_0}^t (\exp(\Lambda(t)))^{-1} W_c^{-1} b(t) dt$  and the integration can be done approximately with an numerical integration method, e.g. [22].

We obtain the solution :

$$\begin{aligned} c(t) &= W_c \exp(\Lambda(t - t_0)) W_c^{-1} c_0 \\ &+ W_c \exp(\Lambda(t)) u(t) , \end{aligned} \quad (13)$$

where  $c_0$  is the initial condition.

*Remark 2.* The solution can be used if we have non-singular matrices, or if the reactants have a successor. Otherwise we obtain fast numerical solvers.

### 3.2 Numerical Methods to ODEs

Here we introduce our numerical methods, which we apply to solve the underlying ODE's for their singular reaction matrix with  $\det(A) = 0$ .

We apply the following methods :

We use the implicit trapezoidal rule

$$\begin{array}{c|cc} 0 & & \\ \hline 1 & \frac{1}{2} & \frac{1}{2} \\ \hline \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{array} \quad (14)$$

Further we use the following implicit Runge-Kutta methods :  
Lobatto IIIA

$$\begin{array}{c|ccc} 0 & 0 & 0 & 0 \\ \frac{1}{2} & \frac{5}{24} & \frac{1}{3} & -\frac{1}{24} \\ 1 & \frac{1}{6} & \frac{2}{3} & -\frac{1}{6} \\ \hline & \frac{1}{6} & \frac{2}{3} & -\frac{1}{6} \end{array} \quad (15)$$

*Remark 3.* We can also apply integration methods for the right hand side.

### 3.3 Numerical Methods to the PDEs

We consider the numerical treatment of the advection equation takes the form,

$$R \phi \partial_t u + \nabla \cdot (\mathbf{v}u) = 0, \quad (t, x) \in [0, T] \times \Omega, \quad (16)$$

$$u(0, x) = U_1(0, x), \quad x \in \partial\Omega, \quad (17)$$

$$u(t, x) = U_2(t, x), \quad (t, x) \in [0, T] \times \partial\Omega^{Dirich}, \quad (18)$$

The initial conditions are given by  $U(0, x)$  and  $u(t, \gamma)$  is explicitly given for  $t > 0$  at the inflow boundary  $\gamma \in \partial^{in}\Omega$ , where  $\partial^{in}\Omega := \{x \in \partial\Omega, \mathbf{n} \cdot \mathbf{v} < 0\}$ . We have  $\partial\Omega^{in} \cup \partial\Omega^{Dirich} = \partial\Omega$ . The exact solution of (16) can be defined directly using the so called “forward tracking” form of characteristics curves. If the solution of (16) is known for some time point  $t_0 \geq 0$  and for some point  $y \in \Omega \cup \partial^{in}\Omega$ , then  $u$  remains constant for  $t \geq t_0$  along the characteristic curve  $X = X(t)$ , i.e.,  $u(t, X(t)) = u(t_0, y)$  and

$$X(t) = X(t; t_0, y) = y + \int_{t_0}^t \frac{\mathbf{v}(X(s))}{R(X(s)) \phi(X(s))} ds. \quad (19)$$

The characteristic curve  $X(t)$  starts at time  $t = t_0$  in the point  $y$ , i.e.  $X(t_0; t_0, y) = y$ , and it is tracked forward in time for  $t > t_0$ . Of course, one can obtain that  $X(t) \notin \Omega$ , i.e., the characteristic curve can leave the domain  $\Omega$  through  $\partial^{out}\Omega$ .

Consequently, one has that  $u(t, X(t; t_0, y)) = U(t_0, y)$ , where the function  $U(0, y)$  is given for  $t_0 = 0$  and  $y \in \Omega$  by initial conditions (16) and for  $t_0 > 0$  and  $y \in \partial^{in}\Omega$  by the inflow boundary conditions (16).

The solution  $u(t, x)$  of (16) can be expressed also in a “backward tracking” form that is more suitable for a direct formulation of discretization schemes. Concretely, for any characteristic curve  $X = X(t) = X(t; s, Y)$  that is defined in a forward manner, i.e.,  $X(s; s, Y) = Y$  and  $t \geq s$ , one obtains the curve  $Y = Y(s) = Y(s; t, x)$  that is defined in a backward manner, i.e.,  $Y(t; t, X) = X$  and  $s \leq t$ . If we express  $Y$  as function of  $t_0$  for  $t_0 \leq t$ , one obtains from (19)

$$Y(t_0) = Y(t_0; t, x) = x - \int_{t_0}^t \frac{\mathbf{v}(X(s))}{R(X(s)) \phi(X(s))} ds \quad (20)$$

and one has  $u(t, x) = u(t_0, Y(t_0))$ .

To simplify our treatment of inflow boundary conditions, we suppose that  $U(t, \gamma) = U^{n+1/2} \equiv \text{const}$  for  $\gamma \in \partial^{in}\Omega$  and  $t \in [t^n, t^{n+1}]$ . Moreover, we define formally for any  $\gamma \in \partial^{in}\Omega$  and  $t_0 \in [t^n, t^{n+1}]$  that  $Y(s; t_0, \gamma) \equiv Y(t_0; t_0, \gamma)$  for  $t^n \leq s \leq t_0$ .

In [12], the so called “flux-based (modified) method of characteristics” was described that can be viewed as an extension of standard finite volume methods (FVMs). The standard FVM for differential equation (16) takes the form

$$|\Omega_j| R_j \phi_j u_j^{n+1} = |\Omega_j| R_j \phi_j u_j^n - \sum_k \int_{t^n}^{t^{n+1}} \int_{\Gamma_{jk}} \mathbf{n}_j(\gamma) \cdot \mathbf{v}(\gamma) u(t, \gamma) d\gamma dt, \quad (21)$$

The idea of flux-based method of characteristics is to apply the substitution  $u(t, \gamma) = u(t^n, Y(t^n; t, \gamma))$  in (21).

Particularly, for the integration variable  $t \in (t^n, t^{n+1})$  and for each point  $\gamma \in \partial^{out}\Omega_j$ , the characteristic curves  $Y(s)$  are tracked backward starting in  $\gamma$  at  $s = t$  and finishing in  $s = t^n$ . One must reach a point  $Y = Y(t^n)$  such that  $Y \in \partial^{in}\Omega$  or  $Y \in \Omega$ . In the first case,  $u(t^n, Y)$  is given by the inflow concentration  $U(t^n, Y) = U^n$ , in the latter one by  $u(t^n, Y)$ .

The integral in right hand side of (21) can be solved exactly for one-dimensional case with general initial and boundary conditions, see e.g. [20]. For general 2D or 3D case, a numerical approximation of  $u(t_0, Y(t_0))$ , respectively of  $Y(t_0)$ , shall be used.

## 4 Splitting methods

The following splitting methods of first order are described. We consider the following ordinary linear differential equations:

$$\partial_t c(t) = A c(t) + B c(t), \quad (22)$$

where the initial-conditions are given as  $c^n = c(t^n)$ . The operators  $A$  and  $B$  are assumed to be bounded linear operators in the Banach-space  $X$  with  $A, B : X \rightarrow X$ . In applications the operators corresponds to the physical operators, e.g. the convection- and the diffusion-operator.

The operator-splitting method is introduced as a method which solve two equation-parts sequentially, with respect to initial conditions. The method is given as following

$$\begin{aligned} \frac{\partial c^*(t)}{\partial t} &= A c^*(t), \quad \text{with } c^*(t^n) = c^n, \\ \frac{\partial c^{**}(t)}{\partial t} &= B c^{**}(t), \quad \text{with } c^{**}(t^n) = c^*(t^{n+1}). \end{aligned} \quad (23)$$

where the time-step is given as  $\tau^n = t^{n+1} - t^n$ . The solution of the equation (22) is  $c^{n+1} = c^{**}(t^{n+1})$ .



The splitting-error of the method is derived with Taylor-expansion, cf. [10]. We obtain the global error as

$$\begin{aligned}\rho_n &= \frac{1}{\tau} (\exp(\tau^n(A+B)) - \exp(\tau^n B) \exp(\tau^n A)) c(t^n) \\ &= \frac{1}{2} \tau^n [A, B] c(t^n) + O((\tau^n)^2) .\end{aligned}\tag{24}$$

where  $[A, B] := AB - BA$  is the commutator of  $A$  and  $B$ . We get an error  $O(\tau^n)$  if the operators  $A$  and  $B$  do not commute, otherwise the method is exact.

#### 4.1 Splitting with respect to the numerical and analytical methods

Here we present a splitting method with respect to the numerical and analytical methods for the differential equations.

Often an analytical method can be used to solve more efficient parts of the full equation system, see [10].

The other part can more efficiently solved by numerical methods.

In our following example, we split a system of ODE's with respect to an analytical method (transformation to an eigenvalue problem) and a numerical method (Trapezoidal rule).

We deal with the following system of ODE's :

$$\frac{dc}{dt} = Mc, \tag{25}$$

$$c(0) = c_0, \tag{26}$$

where  $c = (c_1, \dots, c_n)^t$  is the solution vector of the ODE system.

The reaction matrix  $M$  is given as:

$$M = M_1 + M_2 . \tag{27}$$

where  $M$  is the full matrix of the ODE system and  $M_1$  the part of the analytical method, where  $M_2$  is the part of the numerical method.

So  $\det(M_1) \neq 0$  and  $\det(M_2) = 0$ , that means for the  $M_1$  matrix we can obtain a transformation to an eigenvalue equations, where for  $M_2$  we can not use the transformation to an eigenvalue problem and apply the numerical methods.

Our algorithm is given as:

**Algorithm 1** 1.) Split the reaction matrix  $M : M_1 : \text{Matrix with non zero eigenvalues}, M_2 : \text{Matrix with zero eigenvalues}$

2.) Compute the equation part

$$\frac{d\tilde{c}}{dt} = M_1 \tilde{c}, \tag{28}$$

$$\tilde{c}(0) = \tilde{c}_0, \tag{29}$$

with the analytical method, see section 3.

3.) Compute the equation part

$$\frac{d\hat{c}}{dt} = M_2\hat{c}, \quad (30)$$

$$\hat{c}(0) = \hat{c}_0, \quad (31)$$

with the numerical method, see section 3.

4.) The result is given as :

$$c = (\tilde{c}, \hat{c}).$$

In the next section we discuss the numerical experiments.

## 5 Experiment for the sputter process

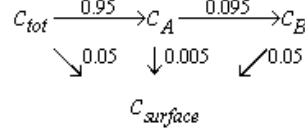
In the following we present the various sputter processes.

### 5.1 Sputter Reactions

In the following experiments we discuss the reaction models of the sputter process.

**Experiment 1: High Energy Level** In this model one assumes a high energy level for the sputtering process. Based on the work of [3].

In the next experiment we deal with the following reaction scheme:



The initial conditions are given with  $c_{tot,0} = 1.0$  and  $c_{A,0} = c_{B,0} = c_{surface,0} = 0$  and we can deal with the following reaction equations:

$$\frac{dc_{tot}}{dt} = -c_{tot}, \quad (32)$$

$$\frac{dc_A}{dt} = 0.95c_{tot} - 0.1c_A, \quad (33)$$

$$\frac{dc_B}{dt} = 0.095c_B - 0.05c_B, \quad (34)$$

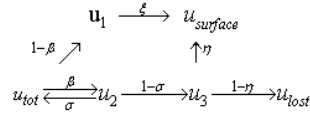
$$\frac{dc_{surface}}{dt} = 0.05c_{tot} + 0.005c_A + 0.05c_B, \quad (35)$$

where the total particle densities is given as  $c_{tot}$ , the single particles are given as  $c_A$  and  $c_B$  and the surface particle density is given as  $c_{surface}$ .

As a result of the computation we show the figure 1. Additionally we present the hysteresis of  $c_a$  and  $c_{surface}$  in Figure 2.

*Remark 4.* The model can be used to have an overview to horizontal gas flows across the thin layer. We can compute the growth rate depending on the amount of the velocity and diffusion. The simulations are done with Maple and Mathematica.

**Experiment 2: Low Energy Sputtering** In the next experiment we deal with a lower energy level and assume the resting of the molecules to a later time at the target layer. For the low energy sputtering we assume a reaction schemes given in [3]:



The initial conditions are given with  $u_{tot,0} = 1.0$  and  $u_{1,0} = u_{2,0} = u_{3,0} = u_{surface,0} = u_{lost} = 0$  and we can deal with the following reaction equations:

$$\frac{du_{tot}}{dt} = -u_{tot} + 0.05u_2, \quad (36)$$

$$\frac{du_1}{dt} = 0.58u_{tot} - 0.9u_1, \quad (37)$$

$$\frac{du_2}{dt} = 0.42u_{tot} - u_2, \quad (38)$$

$$\frac{du_3}{dt} = 0.95u_2 - u_3, \quad (39)$$

$$\frac{du_{surface}}{dt} = 0.9u_1 + 0.4u_3, \quad (40)$$

$$\frac{du_{lost}}{dt} = 0.6u_3. \quad (41)$$

As a result of the computation we get the graphs shown in Figure 3. We present the hysteresis of  $u_1$  and  $u_3$  in Figure 4.

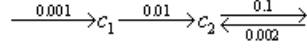
**Experiment 3: Energy level with precursor gas or offsets** In this model one assumes a energy level and a precursor gas for the sputtering process. Based on the work of [3].

In the next experiment we deal with the following reaction:

The initial conditions are given with  $c_{1,0} = 1.0$  and  $c_{2,0} = 0.1$  and we deal with the following reaction equations:

$$\frac{dc_1}{dt} = -0.01c_1 + 0.001, \quad (42)$$

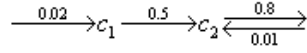
$$\frac{dc_2}{dt} = -0.1c_2 + 0.01c_1 + 0.002. \quad (43)$$



As a result of the computation we get the graphs shown in the Figures 5 and 6.

By changing the reaction coefficients we get the following reaction:

The initial conditions are given with  $c_{1,0} = 1.0$  and  $c_{2,0} = 0.1$  and we get the



following reaction equations:

$$\frac{dc_1}{dt} = -0.5c_1 + 0.02, \quad (44)$$

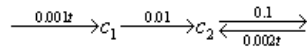
$$\frac{dc_2}{dt} = -0.8c_2 + 0.5c_1 + 0.01. \quad (45)$$

As a result of the computation we get the graphs shown in the figures 7.

*Remark 5.* In the model we assume a homogeneous and inhomogeneous reaction. Because of the small offset of the inhomogeneous reaction, we nearly obtain the same results.

**Experiment 4: Energy level with precursor gas and linear offsets** In this model one assumes an energy level and a precursor gas for the sputtering process. Based on the work of [3]. Here we have a linear offset for the precursor gases.

We can also analyze a variation of this reaction where we have an time dependent inhomogeneous part. The reaction is then given with:



The initial conditions are given with  $c_{1,0} = 1.0$  and  $c_{2,0} = 0.1$  and we can deal with the following reaction equations:

$$\frac{dc_1}{dt} = -0.01c_1 + 0.001t, \quad (46)$$

$$\frac{dc_2}{dt} = -0.1c_2 + 0.002t. \quad (47)$$

As a result of the computation we get the graphs shown in 8.

*Remark 6.* In the model we assume a homogeneous and inhomogeneous reaction. Because of the small offset of the inhomogeneous reaction, we nearly obtain the same results.

## 5.2 Reactive Sputtering Process

A simple model of reactive sputtering described by [1], can describe the understanding of the hysteresis and other properties of the reactive sputtering deposition.

We have the following equations:

$$n_{target} \partial_t \theta_{target} = i \Gamma_r s_r (1 - \theta_{target}) - \Gamma_i \gamma_c \theta_{target} + \quad (48)$$

$$\Gamma_i \gamma_c \theta_{substrate} \quad t \in (0, T), \quad (49)$$

$$n_{substrate} \partial_t \theta_{substrate} = i \Gamma_r s_r (1 - \theta_{substrate}) + \Gamma_i \gamma_c \theta_{target} \frac{A_t}{A_s} - \quad (50)$$

$$\Gamma_i \gamma_m \theta_{substrate} \frac{A_t}{A_s} \quad t \in (0, T), \quad (51)$$

$$\partial_t N_r = \Gamma_r s_r ((1 - \theta_{target}) A_t + (1 - \theta_{substrate}) A_s), \quad (52)$$

$$\Gamma_{sput} = \Gamma_i (\gamma_m \theta_{substrate} + \gamma_c \theta_{target}), \quad (53)$$

where  $\theta_{target}$  and  $\theta_{substrate}$  are the fraction of the target and substrate areas, which is covered by the compound film.  $\gamma_m$  and  $\gamma_c$  be the yields for sputtering the metal and the compound from the target.

We have  $\Gamma_i$  and  $\Gamma_r$  the incident ion and reactive gas molecule fluxes.  $s_r$  the sticking coefficient of a reactive molecule on the metal part of the target. Further  $A_t$  and  $A_s$  are the target and the substrate areas.

The total number of reactive gas molecules per second that are consumed to form the compound deposited on the substrate is  $N_r$ .

The target sputtering flux is  $\Gamma_{sput}$ .

For our experiments we use the following parameters, while we variate the parameters  $\Gamma_r s_r$ ,  $\Gamma_r \gamma_c$ ,  $\Gamma_r \gamma_m$ :

$$\begin{aligned} n_t &= n_s = 1, \\ i &= 1, \\ A_t &= 0.25, \\ A_s &= 0.75, \end{aligned}$$

where the starting conditions are given with  $\theta_{target}, 0 = 1.0$  and  $\theta_{substrate}, 0 = 0.1$ .

**First experiment:** In our first experiment the variable parameters are given with:

$$\begin{aligned} \Gamma_r s_r &= 0.1 \\ \Gamma_r \gamma_c &= 0.07 \\ \Gamma_r \gamma_m &= 0.05 \end{aligned}$$

We get the following simplified system of ODE's:

$$\partial_t \theta_{target} = -0.17\theta_{target} + 0.05\theta_{substrate} + 0.1, \quad t \in (0, T), \quad (54)$$

$$\partial_t \theta_{substrate} = -\frac{0.35}{3}\theta_{substrate} + \frac{0.07}{3}\theta_{target} + 0.1, \quad t \in (0, T), \quad (55)$$

$$\partial_t N_r = -0.025\theta_{target} - 0.075\theta_{substrate} + 0.1, \quad t \in (0, T), \quad (56)$$

$$\Gamma_{sput} = 0.07\theta_{target} + 0.05\theta_{substrate}, \quad t \in (0, T). \quad (57)$$

For the solving of the equations we apply the eigenvalue transformation.

We present the hysteresises of  $\theta_{target}$  with  $\theta_{substrate}$ ,  $\partial_t N_r$  with  $\Gamma_{sput}$  and  $N_r$  with  $\Gamma_{sput}$  in the Figures 9, 10 and 11.

For solving the coupled equations we apply our algorithm 1.

**Second experiment:** In our second experiment the variable parameters are given with:

$$\Gamma_r s_r = 0.05,$$

$$\Gamma_r \gamma_c = 0.02,$$

$$\Gamma_r \gamma_m = 0.1.$$

We get the following simplified system of ODE's:

$$\partial_t \theta_{target} = -0.07\theta_{target} + 0.1\theta_{substrate} + 0.05, \quad t \in (0, T), \quad (58)$$

$$\partial_t \theta_{substrate} = -\frac{0.25}{3}\theta_{substrate} + \frac{0.02}{3}\theta_{target} + 0.05, \quad t \in (0, T), \quad (59)$$

$$\partial_t N_r = -0.0125\theta_{target} - 0.0375\theta_{substrate} + 0.05, \quad t \in (0, T), \quad (60)$$

$$\Gamma_{sput} = 0.02\theta_{target} + 0.1\theta_{substrate}, \quad t \in (0, T). \quad (61)$$

For the solving of the equations we apply the eigenvalue transformation.

We present the hysteresises of  $\theta_{target}$  with  $\theta_{substrate}$ ,  $\partial_t N_r$  with  $\Gamma_{sput}$  and  $N_r$  with  $\Gamma_{sput}$  in the Figures 12, 13 and 14.

For solving the coupled equations we apply our algorithm 1.

**Third experiment:** In our third experiment the variable parameters are given with:

$$\Gamma_r s_r = 0.05,$$

$$\Gamma_r \gamma_c = 0.1,$$

$$\Gamma_r \gamma_m = 0.02.$$

We get the following simplified system of ODE's:

$$\partial_t \theta_{target} = -0.15\theta_{target} + 0.02\theta_{substrate} + 0.05, \quad t \in (0, T), \quad (62)$$

$$\partial_t \theta_{substrate} = -\frac{0.17}{3} \theta_{substrate} + \frac{0.1}{3} \theta_{target} + 0.05, \quad t \in (0, T), \quad (63)$$

$$\partial_t N_r = -0.0125 \theta_{target} - 0.0375 \theta_{substrate} + 0.05, \quad t \in (0, T), \quad (64)$$

$$\Gamma_{sput} = 0.1 \theta_{target} + 0.02 \theta_{substrate}, \quad t \in (0, T). \quad (65)$$

For the solving of the equations we apply the eigenvalue transformation.

We present the hysteresises of  $\theta_{target}$  with  $\theta_{substrate}$ ,  $\partial_t N_r$  with  $\Gamma_{sput}$  and  $N_r$  with  $\Gamma_{sput}$  in the Figures 15, 16 and 17.

For solving the coupled equations we apply our algorithm 1.

### 5.3 Sputtering Process with Convection-Diffusion-Reaction Equation

We can describe our model with continuum equations. Since we want to model the flow very close to the wafer surface, we assume additionally that the flow is dominated by diffusion.

We have the following equation to simulate a first PVD process or sputtering process.

$$\begin{aligned} \partial_t u + v \cdot \nabla u &= u_{in}(t) \text{ in } \Omega \times (0, T), \\ u_0(x, y) &= 0 \text{ on } \Omega, \\ \partial_t u(x, y, 0) &= u_1(x, y) = 0, \\ u(x, y, t) &= 0 \text{ on } \Gamma_1, \\ v \frac{\partial u}{\partial n} &= u_{out} \text{ on } \Gamma_2, \end{aligned} \quad (66)$$

where the constant inflow source is given as  $u_{in}(t)$  and we move the source with respect to the time. So we obtain a sharp step by step processing of different plasma rays. The bottom boundary  $\Gamma_2$  is the outflow boundary, whereas the rest of the boundary  $\Gamma_1$  is a Dirichlet boundary.

We discretize the spatial terms with finite volume methods and apply the BDF method for the time discretization.

Our time steps are given in the Courant number and we apply solver methods that are based on different grid levels, e.g. multi-grid methods.

Figure 18, we present the different phases for the flow field in a simplified sputter process in 2D. To discuss the deposition rates at the metallic bipolar plate, we present in Figure 18 also the growth rate of the flux model in 2D.

*Remark 7.* The model can be used to have an overview to vertical gas flows and the amount of sublimated concentration to the thin film. We can compute the growth rate depending on the outflow of the gas concentration. The simulations are done with Mathematica and numerically with UG.

## 6 Conclusions and Discussions

We present a continuous model, due to a far field and near field idea for a flow field in a PVD apparatus. Based on different models we can predict the flow of

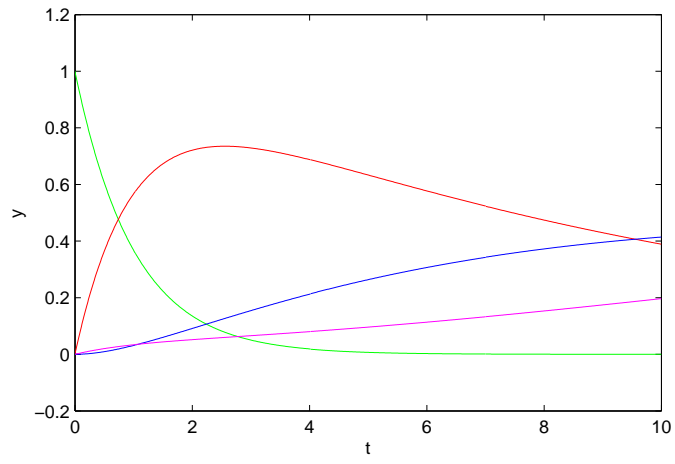
the reacting chemicals on the different scales of the chemical reactor. For the mesoscopic scale model, we discuss the discretization and solver methods. We contribute a coupling algorithm to mix analytical and numerical solutions in our models. Such schemes have benefits in computations and improve the accuracy. Numerical examples are presented to discuss the influence of near-continuum regime at the thin film. The modeling of various inflow sources can describe the growth of the thin-film at the wafer. In future, we will analyze the validity of the models with physical experiments.

## References

1. S. Berg and T. Nyberg. *Fundamental understanding and modeling of reactive sputtering processes*. Thin Solid Films, 476, 215-230, 2005.
2. P. Bastian, K. Birken, K. Eckstein, K. Johannsen, S. Lang, N. Neuss, and H. Rentz-Reichert. *UG - a flexible software toolbox for solving partial differential equations*. Computing and Visualization in Science, 1(1):27-40, 1997.
3. D.J. Christie. *Target material pathways model for high power pulsed magnetron sputtering*. J.Vac.Sci. Technology, 23:2, 330-335, 2005.
4. I. Farago, and Agnes Havasi. *On the convergence and local splitting error of different splitting schemes*. Eötvös Lorand University, Budapest, 2004.
5. P. Csomós, I. Faragó and A. Havasi. *Weighted sequential splittings and their analysis*. Comput. Math. Appl., (to appear)
6. K.-J. Engel, R. Nagel, *One-Parameter Semigroups for Linear Evolution Equations*. Springer, New York, 2000.
7. I. Farago. *Splitting methods for abstract Cauchy problems*. Lect. Notes Comp.Sci. 3401, Springer Verlag, Berlin, 2005, pp. 35-45
8. I. Farago, J. Geiser. *Iterative Operator-Splitting methods for Linear Problems*. Preprint No. 1043 of the Weierstrass Institute for Applied Analysis and Stochastics, Berlin, Germany, June 2005.
9. J. Geiser. *Numerical Simulation of a Model for Transport and Reaction of Radionuclides*. Proceedings of the Large Scale Scientific Computations of Engineering and Environmental Problems, Sozopol, Bulgaria, 2001.
10. J. Geiser. *Gekoppelte Diskretisierungsverfahren für Systeme von Konvektions-Dispersions-Diffusions-Reaktionsgleichungen*. Doktor-Arbeit, Universität Heidelberg, 2003.
11. J. Geiser. *Discretization methods with analytical solutions for convection-diffusion-dispersion-reaction-equations and applications*. Journal of Engineering Mathematics, published online, Oktober 2006.
12. J. Geiser. *Discretisation and Solver Methods with Analytical Methods for Advection-Diffusion-reaction Equations and 2D Applications*. Journal of Porous Media, Begell House Inc., Redding, USA, accepted March, 2008.
13. J. Geiser. *Iterative Operator-Splitting Methods with higher order Time-Integration Methods and Applications for Parabolic Partial Differential Equations*. Journal of Computational and Applied Mathematics, Elsevier, Amsterdam, The Netherlands, 217, 227-242, 2008.
14. J. Geiser. *Decomposition Methods for Partial Differential Equations: Theory and Applications in Multiphysics Problems*. Habilitation Thesis, Humboldt University of Berlin, Germany, under review, July 2008.

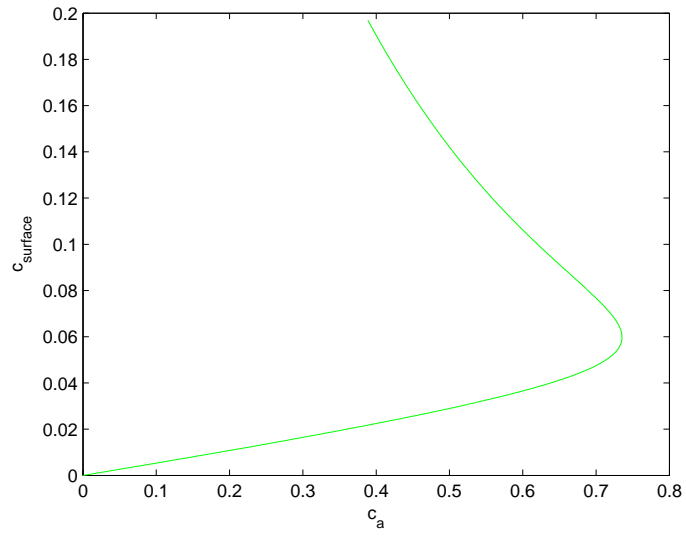


15. M.K. Gobbert and C.A. Ringhofer. *An asymptotic analysis for a model of chemical vapor deposition on a microstructured surface*. SIAM Journal on Applied Mathematics, 58, 737–752, 1998.
16. H.H. Lee. *Fundamentals of Microelectronics Processing* McGraw-Hill, New York, 1990.
17. Chr. Lubich. *A variational splitting integrator for quantum molecular dynamics*. Report, 2003.
18. S. Middleman and A.K. Hochberg. *Process Engineering Analysis in Semiconductor Device Fabrication* McGraw-Hill, New York, 1993.
19. M. Ohring. *Materials Science of Thin Films*. Academic Press, San Diego, New York, Boston, London, Second edition, 2002.
20. P.J. Roache. *A flux-based modified method of characteristics*. Int. J. Numer. Methods Fluids, 12:12591275, 1992.
21. T.K. Senega and R.P. Brinkmann. *A multi-component transport model for non-equilibrium low-temperature low-pressure plasmas*. J. Phys. D: Appl.Phys., 39, 1606–1618, 2006.
22. J. Stoer and R. Burlisch. *Introduction to numerical analysis*. Springer verlag, New York, 1993.

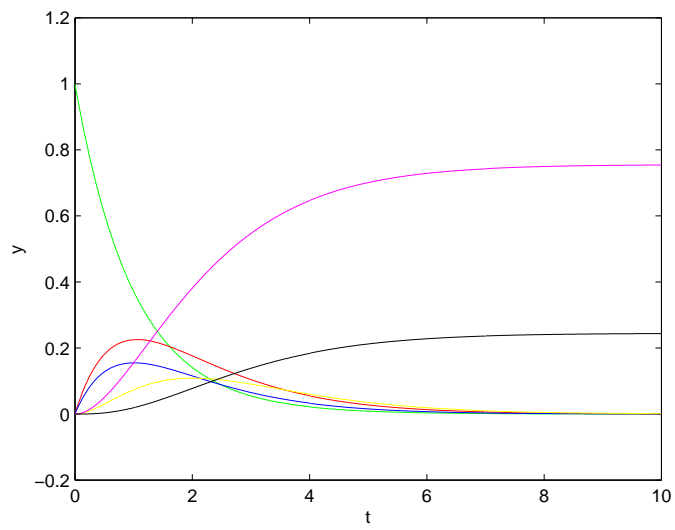


**Fig. 1.** 1D experiment of the heavy particle transport.

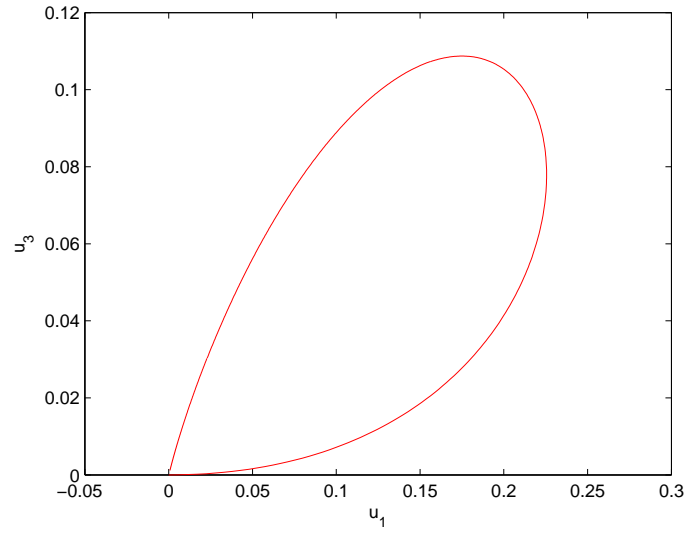
green:  $c_{tot}$ , red:  $c_A$ , blue:  $c_B$ , magenta:  $c_{surface}$



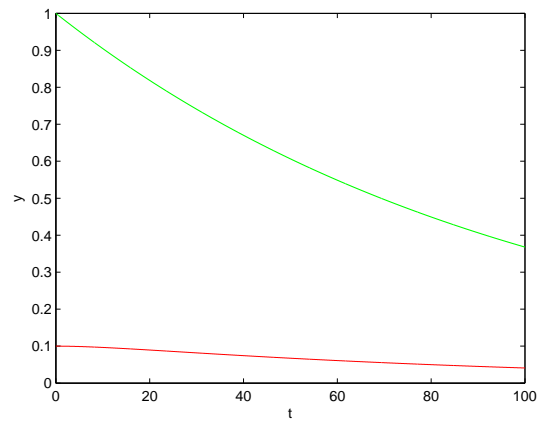
**Fig. 2.** hysteresis of the concentrations  $c_a$  and  $c_{surface}$



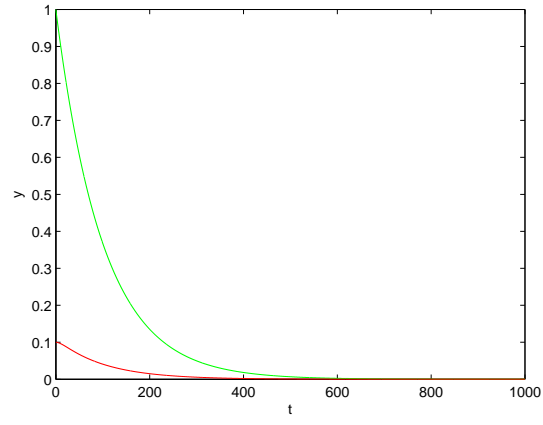
**Fig. 3.** 1D experiment of the heavy particle transport.  
green:  $u_{tot}$ , red:  $u_1$ , blue:  $u_2$ , magenta:  $u_{surface}$ , black:  $u_{lost}$



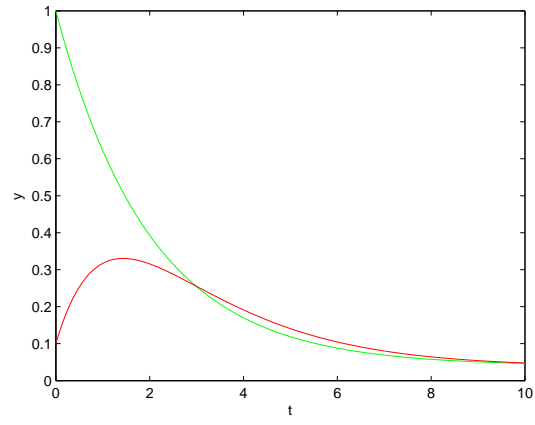
**Fig. 4.** hysteresis of the concentrations  $u_1$  and  $u_3$



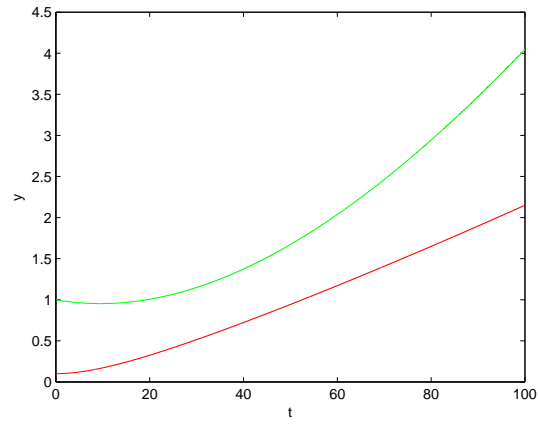
**Fig. 5.** 1D experiment of the heavy particle transport. green:  $c_1$ , red:  $c_2$



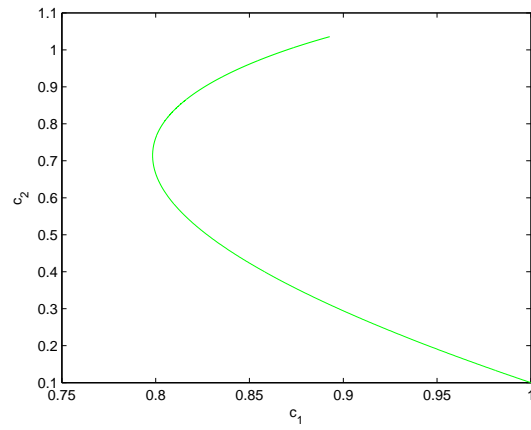
**Fig. 6.** 1D experiment of the heavy particle transport.green:  $c_1$ , red:  $c_2$



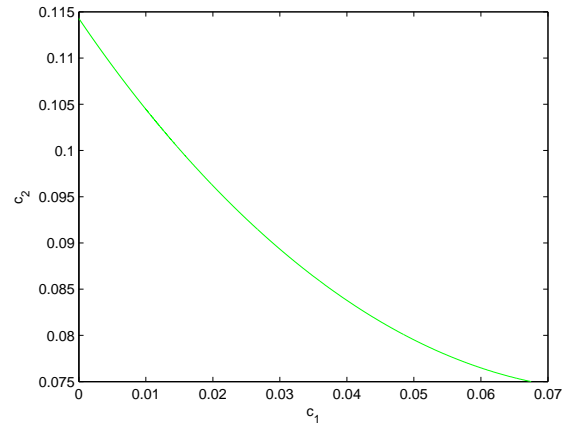
**Fig. 7.** 1D experiment of the heavy particle transport.green:  $c_1$ , red:  $c_2$



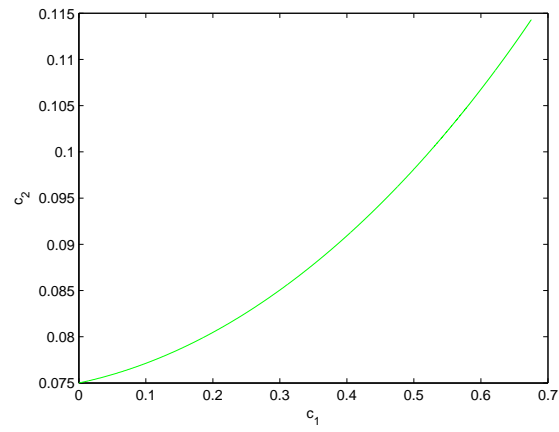
**Fig. 8.** 1D experiment of the heavy particle transport. green:  $c_1$ , red:  $c_2$



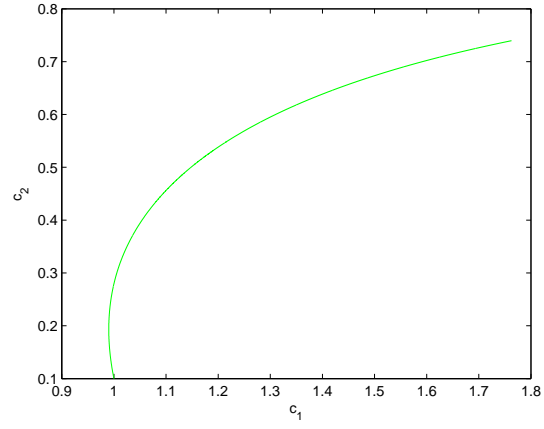
**Fig. 9.** first experiment: hysteresis of  $\theta_{target}$  ( $c_1$ ) with  $\theta_{substrate}$  ( $c_2$ )



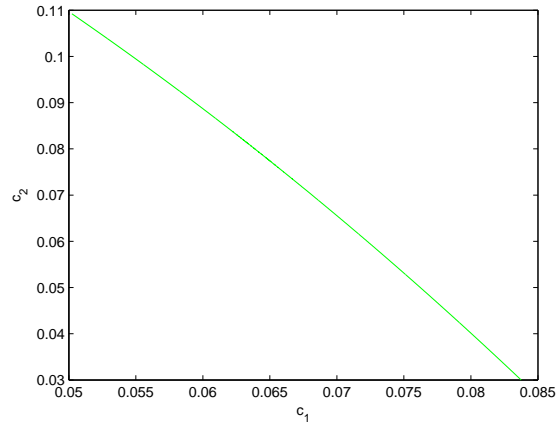
**Fig. 10.** first experiment: hysteresis of  $\partial_t N_r(c_1)$  with  $\Gamma_{sput}(c_2)$



**Fig. 11.** first experiment: hysteresis of  $N_r(c_1)$  with  $\Gamma_{sput}(c_2)$

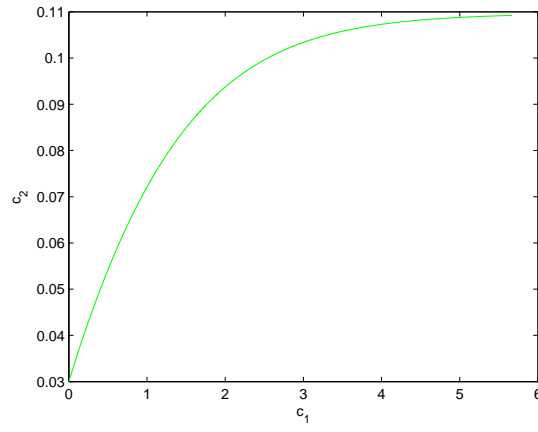


**Fig. 12.** second experiment: hysteresis of  $\theta_{target}$  ( $c_1$ ) with  $\theta_{substrate}$  ( $c_2$ )

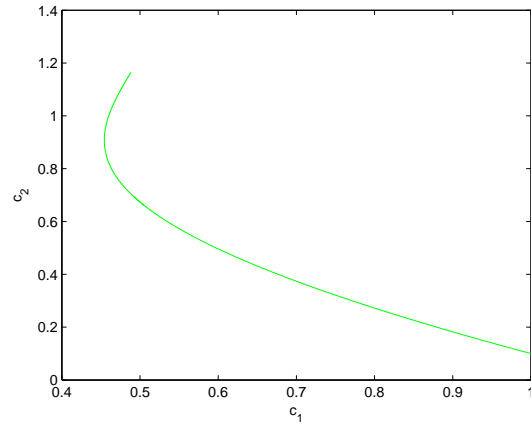


**Fig. 13.** second experiment: hysteresis of  $\partial_t N_r$  ( $c_1$ ) with  $\Gamma_{sput}$  ( $c_2$ )

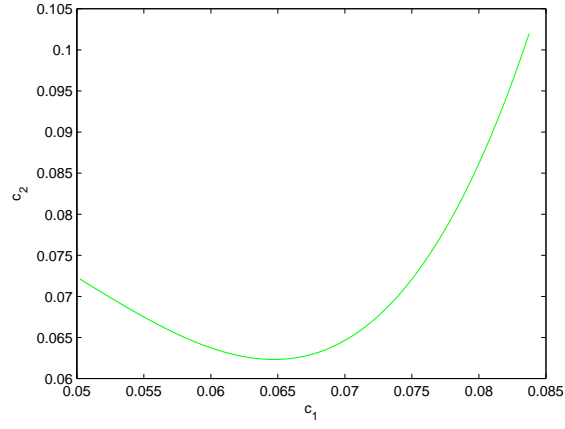




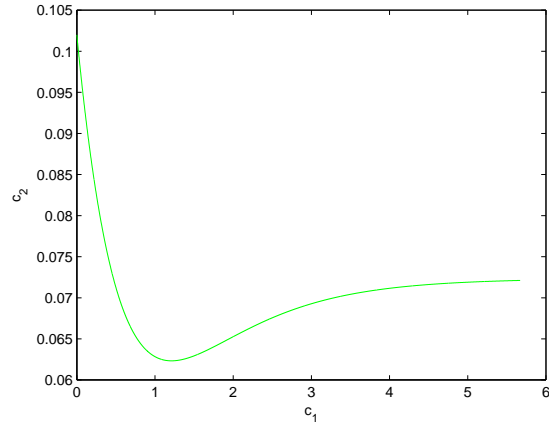
**Fig. 14.** second experiment: hysteresis of  $N_r (c_1)$  with  $\Gamma_{sput} (c_2)$



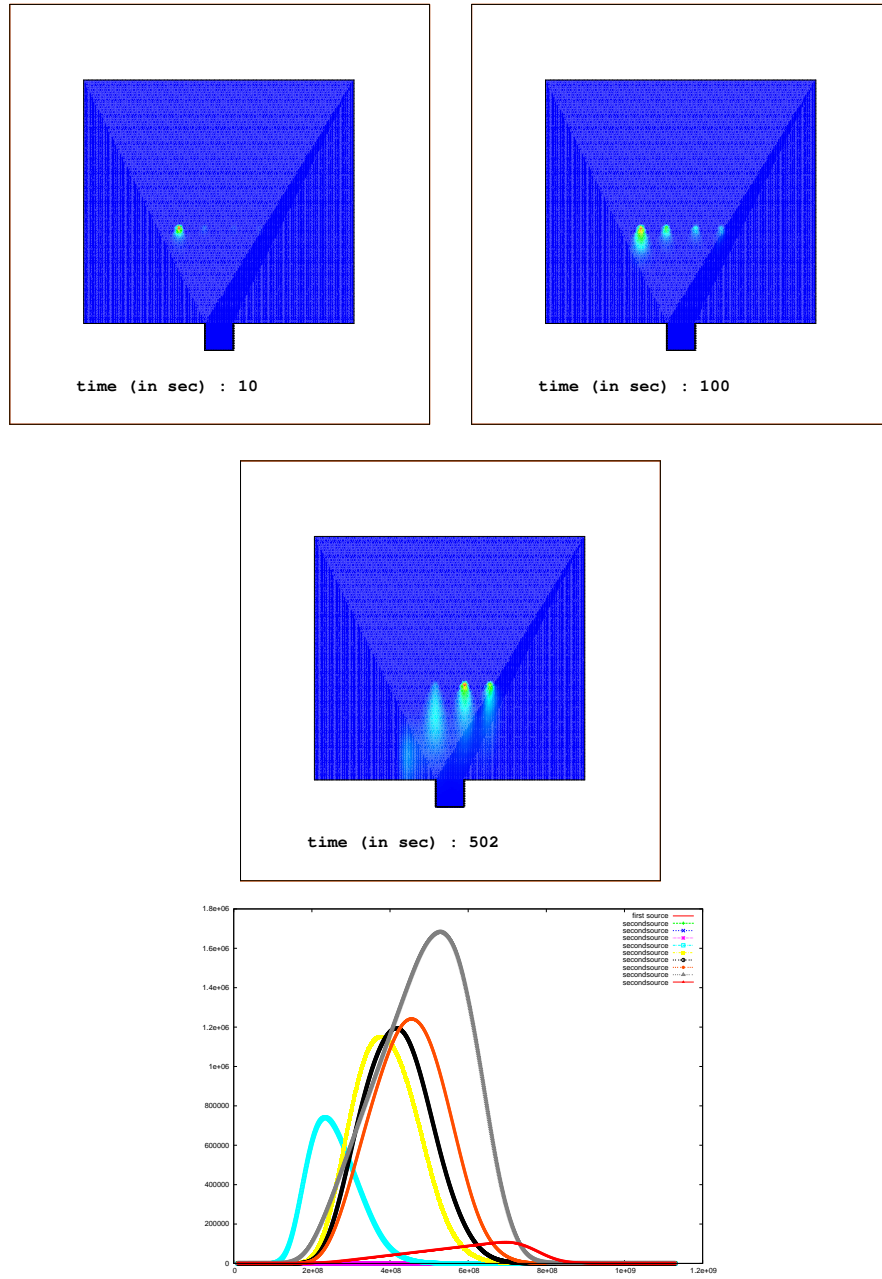
**Fig. 15.** third experiment: hysteresis of  $\theta_{target} (c_1)$  with  $\theta_{substrate} (c_2)$



**Fig. 16.** third experiment: hysteresis of  $\partial_t N_r(c_1)$  with  $\Gamma_{sput}(c_2)$



**Fig. 17.** third experiment: hysteresis of  $N_r(c_1)$  with  $\Gamma_{sput}(c_2)$



**Fig. 18.** 2D experiment of the apparatus with a sputtering source at 3 different time levels (Initial, Start and Flux Positions) and growth of the thin film at the wafer surface

## 7 Appendix

### 7.1 Matlab-Code for experiment 1

```

put
A = [-1,0,0; 0.95,-0.1,0; 0,0.095,-0.05 ];
yo = [1; 0; 0];
to = 0.0;
tf = 10.0;
n = max(size(A));
% compute eigenvalues and eigenvectors
[wcol, lambdac] = eig(A);
wcolinv = inv(wcol);
% set up discretized solution grid
npoints = 1000;
c0 = 0;
dt = (tf-to)/npoints;
for i = 1:1:npoints+1
    tp(i) = (i-1)*dt + to;
end
% calculate solution
explambda = zeros(n,n);
yp = zeros(n,npoints);
cp = zeros(1,npoints);
for i = 1:1:npoints+1
    for j = 1:n
        explambda(j,j) = exp(lambdac(j,j)*(tp(i) - to ));
    end
    yp(:,i) = wcol*explambda*wcolinv*yo;
    if(i,1)
        cp(1,i) = c0 + (1/2)*dt*0.05*y1 + (1/2)*dt*0.05*yp(1,i) + (1/2)*dt*0.005*y2
        + (1/2)*dt*0.005*yp(2,i) + (1/2)*dt*0.05*y3 + (1/2)*dt*0.05*yp(3,i);
        c0 = cp(1,i);
    end
    y1 = yp(1,i);
    y2 = yp(2,i);
    y3 = yp(3,i);
end
%plot
for j = 1:1:4
    if (j==1)
        plot (tp,yp(j,:), 'g-'), xlabel( 't' ), ylabel ( 'y' )
    elseif (j==2)
        plot (tp,yp(j,:), 'r-'), xlabel( 't' ), ylabel ( 'y' )
    elseif (j==3)
        plot (tp,yp(j,:), 'b-'), xlabel( 't' ), ylabel ( 'y' )
    end
end

```

```

elseif (j==4)
plot (tp,cp(1,:), 'm-'), xlabel( 't' ), ylabel ( 'y' )
end
hold on
end
hold off

```

## 7.2 Matlab-Code for experiment 2

```

put
A = [ -1, 0, 0.05, 0; 0.58, -0.9, 0, 0; 0.42, 0, -1, 0; 0, 0, 0.95, -1 ];
yo = [1;0;0;0];
to = 0.0;
tf = 10.0;
n = max(size(A));
c0=0;
cl0=0;
% compute eigenvalues and eigenvectors
wcol,lambdac
= eig(A);
wcolinv = inv(wcol);
% set up discretized solution grid
npoints = 1000;
dt = (tf-to)/npoints;
for i = 1:1:npoints+1
tp(i) = (i-1)*dt +to;
end
% calculate solution
explambda = zeros(n,n);
yp = zeros(n,npoints);
cp = zeros(1,npoints);
cl = zeros(1,npoints);
for i = 1:1:npoints+1
for j = 1:n
explambda(j,j) = exp(lambdac(j,j)*(tp(i) - to) );
end
yp(:,i) = wcol*explambda*wcolinv*yo;
if(i>1)
cp(1,i)=c0 + (1/2)*dt*0.9*u1 + (1/2)*dt*0.9*yp(2,i) + (1/2)*dt*0.4*u3 + (1/2)*dt*0.4*yp(4,i);
c0=cp(1,i);
cl(1,i)=cl0 + (1/2)*dt*0.6*u3 + (1/2)*dt*0.6*yp(4,i);
cl0=cl(1,i);
end
u1=yp(2,i);
u3=yp(4,i);
end

```

```

% plot
for j = 1:1:(n+2)
if (j==1)
plot (tp,yp(j,:), 'g-'), xlabel( 't' ), ylabel ( 'y' )
elseif (j==2)
plot (tp,yp(j,:), 'r-'), xlabel( 't' ), ylabel ( 'y' )
elseif (j==3)
plot (tp,yp(j,:), 'b-'), xlabel( 't' ), ylabel ( 'y' )
elseif (j==4)
plot (tp,yp(j,:), 'y-'), xlabel( 't' ), ylabel ( 'y' )
elseif (j==5)
plot (tp,cp(1,:), 'm-'), xlabel( 't' ), ylabel ( 'y' )
elseif (j==6)
plot (tp,cl(1,:), 'k'), xlabel( 't' ), ylabel ( 'y' )
end
hold on
end
hold off

```

### 7.3 Matlab-Code for experiment 3

```

put
A = [ -0.01, 0; 0.01, -0.1];
yo = [1; 0.1];
to = 0.0;
tf = 10.0;
n = max(size(A));
b = [0.001; 0.002];
% compute eigenvalues and eigenvectors
wcol, lambdac
= eig(A);
wcolinv = inv(wcol);
% set up discretized solution grid
npoints = 1000;
dt = (tf-to)/npoints;
for i = 1:1:npoints+1
tp(i) = (i-1)*dt + to;
end
% calculate solution
explambda = zeros(n,n);
explambda2 = zeros(n,n);
yp = zeros(n,npoints);
u = zeros(n,npoints);
tlast = 0;
for i = 1:1:npoints+1
for j = 1:n

```

```

explambda(j,j) = exp( lambdac(j,j)*(tp(i) - to) );
explambda1(j,j) = exp(-lambdac(j,j)*tlast);
explambda2(j,j) = exp(-lambdac(j,j)*tp(i));
end
if(i greater 1)
u(:,i) = u1 + (1/2)*dt*explambda1*wcolinv*b(:,1) + (1/2)*dt*explambda2*wcolinv*b(:,1);
yp(:,i) = wcol*explambda*wcolinv*yo + wcol*exp(lambdac(j,j)*tp(i))*u(:,i);
end
u1 = u(:,i);
tlast = tp(i);
end
% plot
for j = 1:1:n
if (j==1)
plot (tp,yp(j,:), 'g-'), xlabel( 't' ), ylabel ( 'y' )
elseif (j==2)
plot (tp,yp(j,:), 'r-'), xlabel( 't' ), ylabel ( 'y' )
end
hold on
end
hold off

```

#### 7.4 Matlab-Code for experiment 4

```

put
A = [ -0.01, 0; 0.01, -0.1];
yo = [1; 0.1];
to = 0.0;
tf = 10.0;
n = max(size(A));
b = [0.001; 0.002];
% compute eigenvalues and eigenvectors
wcol,lambdac
= eig(A);
wcolinv = inv(wcol);
% set up discretized solution grid
npoints = 1000;
dt = (tf-to)/npoints;
for i = 1:1:npoints+1
tp(i) = (i-1)*dt +to;
end
% calculate solution
explambda = zeros(n,n);
explambda2 = zeros(n,n);
yp = zeros(n,npoints);
u = zeros(n,npoints);

```

```

tlast = 0;
for i = 1:1:npoints+1
for j = 1:n
explambda(j,j) = exp( lambdac(j,j)*(tp(i) - to) );
explambda1(j,j) = exp(-lambdac(j,j)*tlast);
explambda2(j,j) = exp(-lambdac(j,j)*tp(i));
end
if(i greater than 1)
u(:,i) = u1 + (1/2)*dt*explambda1*wcolinv*(b(:,1)*(tlast))
+ (1/2)*dt*explambda2*wcolinv*(b(:,1)*(tp(i)));
yp(:,i) = wcol*explambda*wcolinv*yo + wcol*exp(lambdac(j,j)*tp(i))*u(:,i);
end
u1 = u(:,i);
tlast = tp(i);
end
%plot
for j = 1:1:n
if (j==1)
plot (tp,yp(j,:), 'g-'), xlabel( 't' ), ylabel ( 'y' )
elseif (j==2)
plot (tp,yp(j,:), 'r-'), xlabel( 't' ), ylabel ( 'y' )
end
hold on
end
hold off

```